CLAIMS

Having thus described our invention, what we claim as new, and desire to secure by Letters Patent is:

1. A simulator of intelligent workstations at level 2 of an OSI model for generating one or more unique network data frames for transmission, the simulator comprising:

one or more protocol application modules for generating one or more data streams that emulate one or more client requests from one or more client workstations;

one or more protocol stacks each representing a group of protocols at multiple layers, said one or more protocol stacks coupled to said one or more protocol application modules and receiving said one or more data streams to encapsulate said one or more data streams with associated one or more protocol stacks; and

a data delivery module for receiving one or more encapsulated data streams and inserting a unique address identifier into each of said one or more encapsulated data streams to generate one or more network data frames, each inserted unique address identifier corresponding to an emulated client workstation that originated the client request associated with said one or more encapsulated data streams.

2. The simulator of intelligent workstation at level 2 as claimed in claim 1, wherein the simulator further includes:

an insertion module for placing said one or more generated network data frames onto a communications medium for delivery to a system under test.

3. The simulator of intelligent workstations at level 2 as claimed in claim 1, the simulator further including:

a scripting interface module coupled to said one or more protocol application modules, the scripting interface module receiving one or more emulated user actions to interpret and pass said user actions to said one or more protocol application modules for generating said one or more data streams.

4. The simulator of intelligent workstations at level 2 as claimed in claim 3, wherein the scripting interface module includes:

one or more scripts having one or more instructions for emulating user actions, wherein said emulated user actions are received from the script.

5. The simulator of intelligent workstations at level 2 as claimed in claim 3, wherein the scripting interface module further includes:

a service module that interprets said one or more instructions for emulating user actions.

1   6. The simulator of intelligent workstations

2   at level 2 as claimed in claim 3, wherein the simulator

3   further includes:

4   a command and control service module coupled to

5   said scripting interface module for enabling dynamic

6   loading and execution of said one or more instructions

7   emulating user actions.

1   7. The simulator of intelligent workstations

2   at level 2 as claimed in claim 1, the simulator further

3   including:

4   a command and control service module coupled to

5   said one or more protocol stacks for enabling dynamic

6   loading of said one or more protocol stacks.

1   8. The simulator of intelligent workstations

2   at level 2 as claimed in claim 1, the simulator further

3   including:

4   an application programming interface interposed

5   between said one or more protocol stacks and the data

6   delivery module, said application programming interface

7   enabling each one of said one or more protocol stacks to

8   communicate with the data delivery module without

9   modifications to any one of said one or more protocol

10   stacks and the data delivery module.

1   9. The simulator of intelligent workstations

2   at level 2 as claimed in claim 1, the simulator further

3   including:

4   an application programming interface interposed

5 between said one or more protocol application modules and

6 said one or more protocol stacks, said application

7 programming interface enabling each one of said one or

8 more protocol application modules to communicate with

9 each one of said one or more protocol stacks without

10 modifications to any one of said one or more protocol

11 application modules and said one or more protocol stacks.

1   10. The simulator of intelligent workstations

2 at level 2 as claimed in claim 3, the simulator further

3 including:

4   an application programming interface interposed

5 between the scripting interface module and said one or

6 more protocol application modules, said application

7 programming interface enabling the scripting interface

8 module to communicate with each one of said one or more

9 protocol application modules without modifications to any

10 one of said one or more protocol application modules and

11 the scripting interface module.

1   11. The simulator of intelligent workstations

2 at level 2 as claimed in claim 1, wherein said one or

3 more protocol stacks is associated with a protocol suite

4 selected from the group comprising:

5   a TCP/IP protocol suite;

6   a UDP/IP protocol suite; and

7   an SSL suite.

12. The simulator of intelligent workstations at level 2 as claimed in claim 2, wherein said communications medium includes LAN.

13. The simulator of intelligent workstations at level 2 as claimed in claim 2, wherein said communications medium includes an input/output (I/O) buffer internal to a system under test for servicing said one or more emulated client requests.

14. The simulator of intelligent workstations at level 2 as claimed in claim 1, wherein said network data frames are transmitted to a system under test, said simulator is further enabled to receive server responses to said client requests and check said system under test responses to ensure correct system under test operation.

15. The simulator of intelligent workstations at level 2 as claimed in claim 1, wherein said simulator is enabled to capture and maintain client state data from a system under test and facilitate client transmissions using said state data.

16. A method for generating network data frames for simulation of a plurality of intelligent workstations at level 2 of a protocol stack, the method comprising:
generating one or more data streams representing one or more emulated client requests from one or more workstations identifier;

8   inserting header data associated with a

9   selected protocol stack into said one or more data

10   streams to generate one or more protocol encapsulated

11   data frames; and

12   adding the unique address identifier to said

13   one or more protocol encapsulated data frames to generate

14   network data frames for transmission to a system under

15   test.

1   17.   The method for generating network data

2   frames as claimed in claim 16, the method further

3   including:

4   receiving user actions for initiating emulation

5   of said client requests.

1   18.   The method for generating a network data

2   frames as claimed in claim 16, the method further

3   including:

4   placing said generated network data frames onto

5   a communications medium.

1   19.   The method for generating a network data

2   frames as claimed in claim 16, the method further

3   including:

4   placing said generated network data frames into

5   an input/output (I/O) buffer for a system under test to

6   receive and respond to said client requests represented

7   by said generated network data frames.

1　　　　　20. The method for generating a generalized

2　network data frames as claimed in claim 16, the method

3　further including:

4　　　　　dynamically loading said selected protocol

5　stack for generating said one or more protocol

6　encapsulated data frames.

1　　　　　21. The method for generating a network data

2　frames as claimed in claim 16, the method further

3　including:

4　　　　　receiving responses associated with said client

5　requests from the system under test and checking said

6　responses to ensure correct operation of system under

7　test.

1　　　　　22　The method for generating a network data

2　frames as claimed in claim 16, the method further

3　including:

4　　　　　capturing and maintaining client state data

5　from the system under test and facilitating transmissions

6　using said state data.

1　　　　　23. The method for generating a network data

2　frames as claimed in claim 17, the method further

3　including:

4　　　　　formulating said user actions into a plurality

5　of transactions for communication to and from a unit

6　under test, wherein said one or more data streams

7　representing emulated client requests are generated from

8　said plurality of transactions.

1   24. A program storage device readable by
2 machine, tangibly embodying a program of instructions
3 executable by the machine to perform the method steps of
4 generating network data frames for simulation of a
5 plurality of intelligent workstations at level 2 of a
6 protocol stack, the method steps comprising:
7   generating one or more data streams
8 representing one or more emulated client requests from
9 one or more workstations;
10   inserting header data associated with a
11 selected protocol stack into said one or more data
12 streams to generate one or more protocol encapsulated
13 data frames; and
14   adding a unique address identifier to said one
15 or more protocol encapsulated data frames to generate
16 network data frames for transmission, each inserted
17 unique address identifier corresponding to an emulated
18 client workstation that originated the client request
19 associated with said one or more encapsulated data
20 frames.

1   25. The program storage as claimed in claim
2 24, the method steps further comprising:
3   receiving user actions for initiating emulation
4 of said client requests.

1   26. The program storage as claimed in claim
2 24, the method steps further comprising:

3   placing said generated network data frames onto

4   a communications medium.